

# OOOS: A hardware-independent SLR control system

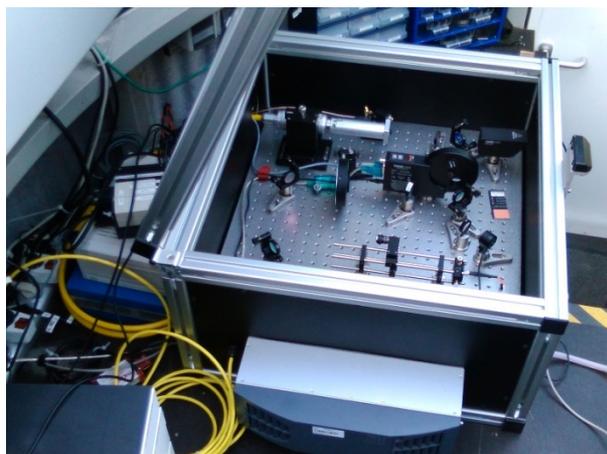
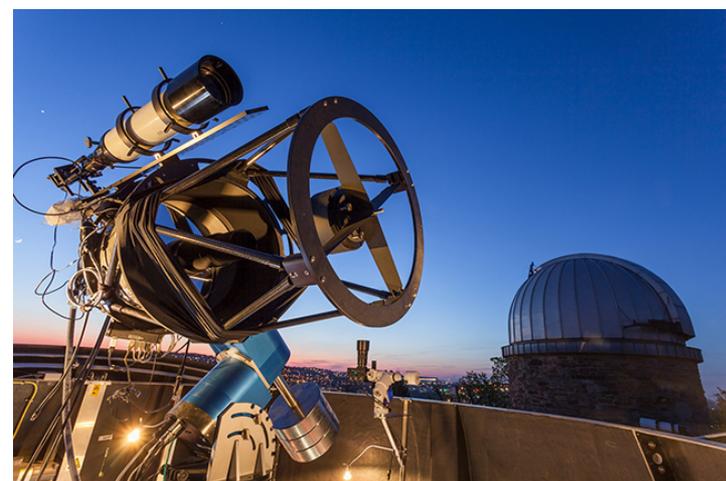
Daniel Hampf, Fabian Sproll, et. al.  
German Aerospace Center, Stuttgart



Knowledge for Tomorrow

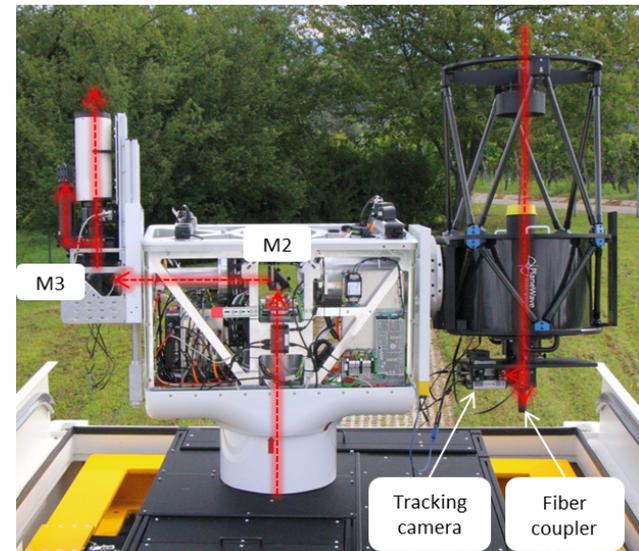
# Stuttgart SLR Systems (I): UFO / UROL

- Joined the ILRS in August 2017 (engineering station)
- Main features:
  - Small footprint / low cost
  - Fibre-transmission of laser pulses (no coudé path)
  - IR-Ranging (1064 nm)
  - High repetition ranging ( $\sim 100$  kHz)



# Stuttgart SLR Systems (II): STAR-C

- Currently in set-up phase near Stuttgart
- Main features:
  - Transportable (single container)
  - coudé path, strong laser
  - Main application: Space Debris Ranging



# OOOS (Orbital Objects Observation Software)

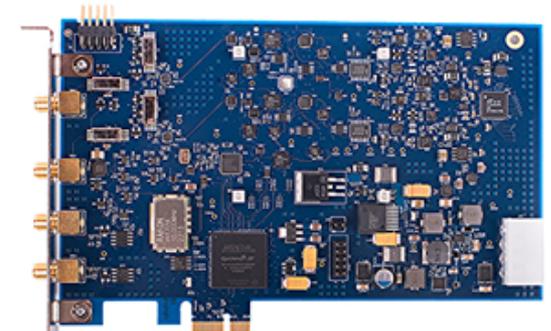
- Why hardware-independent?
  - Is used on UFO and on STAR-C
  - As engineering station, we change hardware quite a lot
- Why automatization:
  - Sometimes we need to gather data for verification  
-- but we have no personnel for routine observations
  - STAR-C is envisaged to operate autonomously at remote sites



*Riga A033-ET*



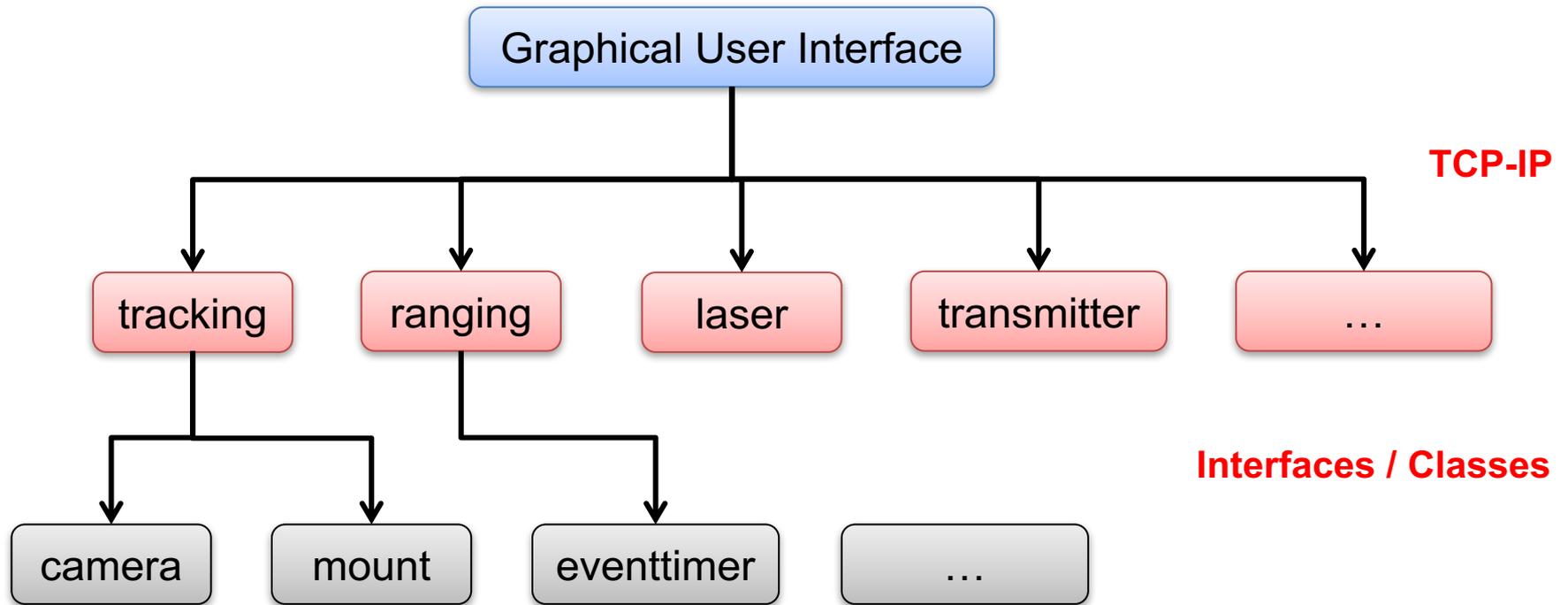
*PicoQuant HydraHarp 400*



*Guidetech GT668-SLR*



# Modular 3-layer design



- Andor Zyla
- Point Grey
- Canon DSLR
- ...



# Streamlined User Interface

The screenshot displays the OODS (Observation and Observation Data System) software interface. The main window is titled 'OODS' and contains several panels:

- Telescope Configuration:**
  - Tracking: 1.58 arcsec
  - Laser: 154.097 mW (Laser source), 000.000 mW (Laser transmitter)
  - Ranging: 1000 m (Signal start), 1700 m (Signal return), 100 m (Signal in window)
  - Object: 757 seconds to stop (Target object)
- Tracking Plot:** A scatter plot showing 'Rangefinder (µm)' on the y-axis (ranging from 2.9 to 3.7) versus 'Time (s)' on the x-axis (ranging from 260 to 300). A blue line indicates a downward trend in rangefinder values over time.
- RA / DEC offsets:** A control panel with directional arrows and a 'Reset' button. Current offsets are RA: -0.1717" and DE: +0.0328". Step size is 368 arcsec (0.302").
- Camera presets:** A table showing different camera configurations:

ROI x	ROI y	Binning
2560	2160	3
1000	1000	2
800	800	2
800	800	1

- Laser control:** A panel with 'Filter' set to '1 (none)' and 'Shutter' set to 'open'. It includes 'Previous', 'Next', 'Open', and 'Close' buttons.
- Log Console:** A scrollable area at the bottom left showing system messages such as 'Telescope message: INFO DEC:23 Device motion state changed'.
- Status Bar:** Shows 'TRACKING' and 'Mount time: 17:54:52'.



The screenshot displays the OOOS software interface, which is divided into several sections:

- Menu Bar:** OOOS View Planning Hardware Control SkyView Calibration
- Sub-Menu Bar:** Planning Hardware Control Cameras SkyView Results DomeView
- Sky Chart (Left):** A circular sky chart with concentric circles and a central cross. The cardinal directions are labeled: N (North) at the top, S (South) at the bottom, W (West) on the left, and E (East) on the right. A yellow dot is positioned in the lower-left quadrant, labeled "LAGEOS 1". A blue arc is drawn in the upper-left quadrant.
- All-Sky Camera View (Right):** A grayscale image of the Moon's surface, showing a bright horizon and a dark sky. The title bar for this view is "All-Sky Camera" and the file path is ".../temp/allsky\_image.fit".
- Camera Controls (Bottom Right):** A panel with various settings:
  - Max. Int: 65535
  - Stretch: 1,000% (with up/down arrows)
  - Exposure: 0.1 (with a green slider)
  - ROI x: 100 (with a green slider)
  - Binning: 2 (with a green slider)
  - ROI y: 100 (with a green slider)
  - Max framerate: 1.0 (with a green slider)
  - Automatic mode:
  - Cur framerate: 1.00 Hz
  - Camera presets button
  - Coordinates: (-265, 521): 2483
  - Play button
- Log Console (Bottom Left):** A text area showing system messages:
  - 12:56:15: WARNING: Some TLE data sets are older than 7 days
  - 12:56:15: Loaded TLE data for 69 objects
  - 12:55:25: WARNING: Some TLE data sets are older than 7 days
  - 12:55:25: Loaded TLE data for 69 objects
  - 12:54:05: All selected daemons have been connected.
  - 12:54:05: Success
- System Time (Bottom Right):** System time: 12:56:50



# Implementation

- Pure python (but using C libraries)
  - Simple installation
  - Quick debugging
- Cross-platform
- No real-time requirements (all real-time operations are done on hardware)
- Designed from scratch in 2016, no legacy code
- Multiprocessing to handle large data rates (“100 kHz ranging”)
- Loose coupling between modules

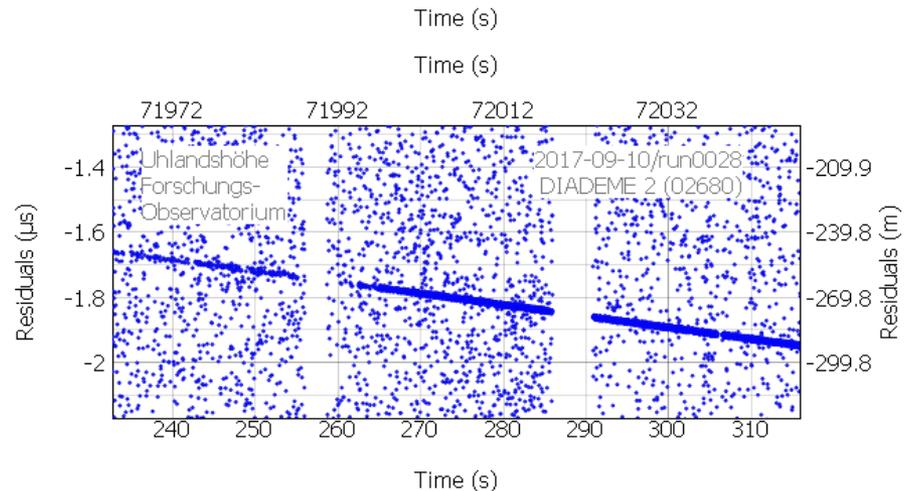
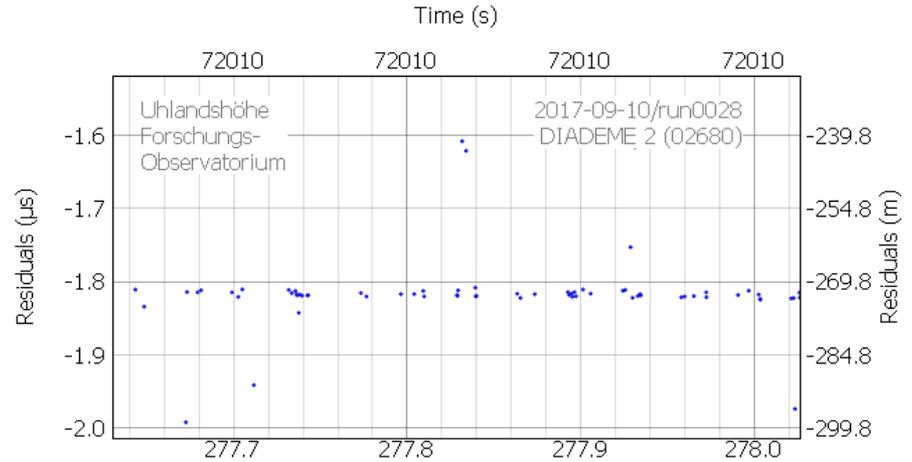
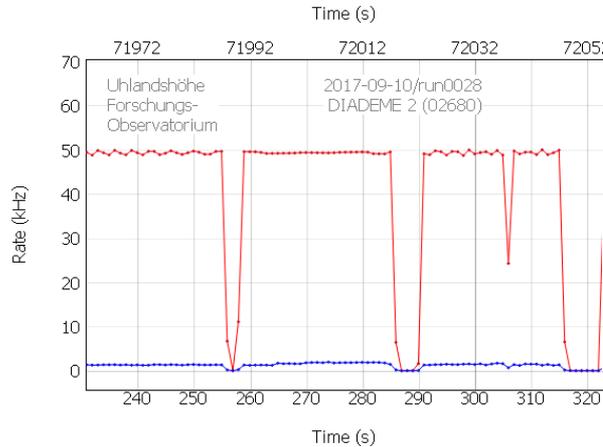
```
runnumber = 15
target_type = sat
target_name = ENVISAT
target_ID = 27386
target_RCS = 18.597
ILRS_name = ENVISAT
TLE1 = 1 27386U 02009A 17241.38234473 .00000005 00000-0 15333-4 0 9990
TLE2 = 2 27386 98.2252 283.5801 0001322 79.3709 280.7627 14.37901661811590
CPF_filename = Z:/data\2017-08-29\run0015\info/CPF_prediction1.txt
```

*Message GUI → tracking daemon*



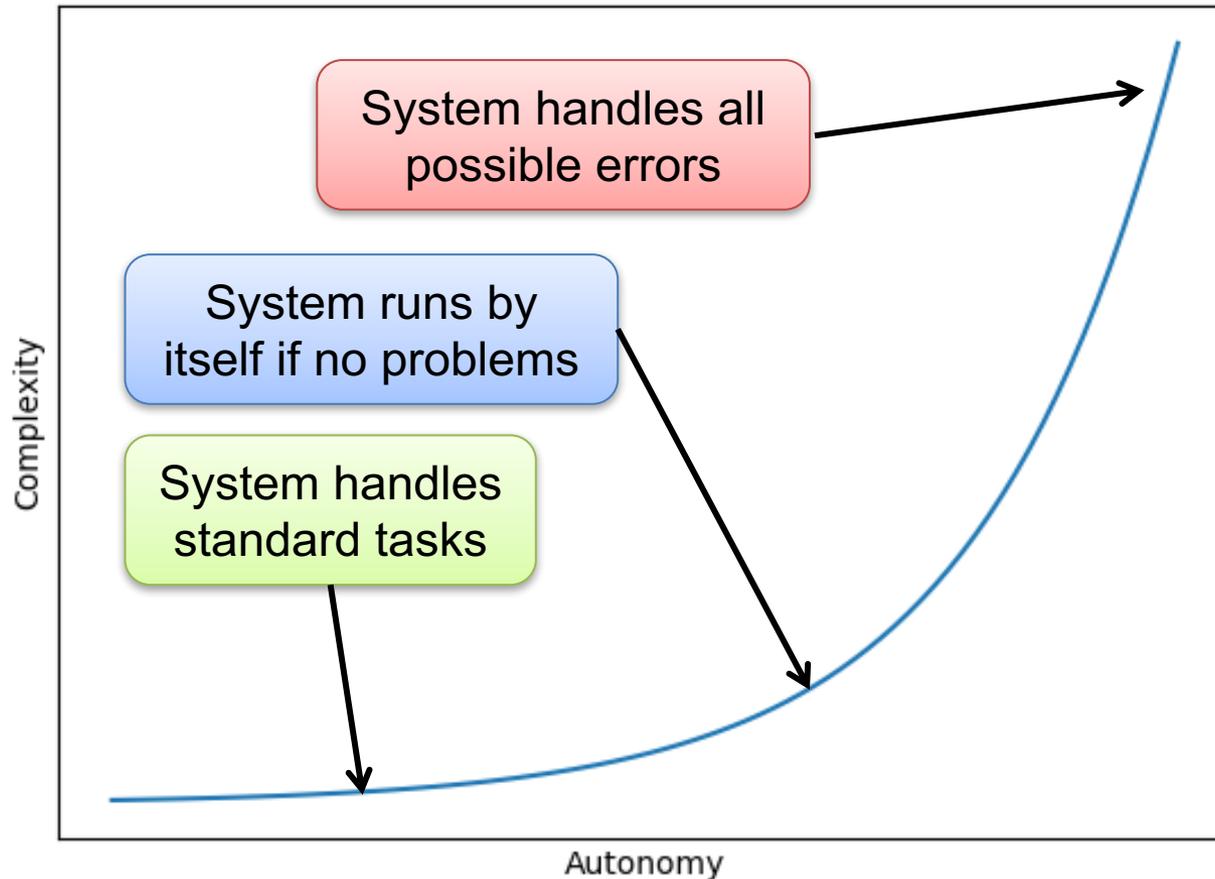
# 100 kHz Ranging

- Why?
  - Fibre transmitter can tolerate only low pulse energy
  - More statistics give better accuracy
- How?
  - Burst mode to avoid pulse collisions



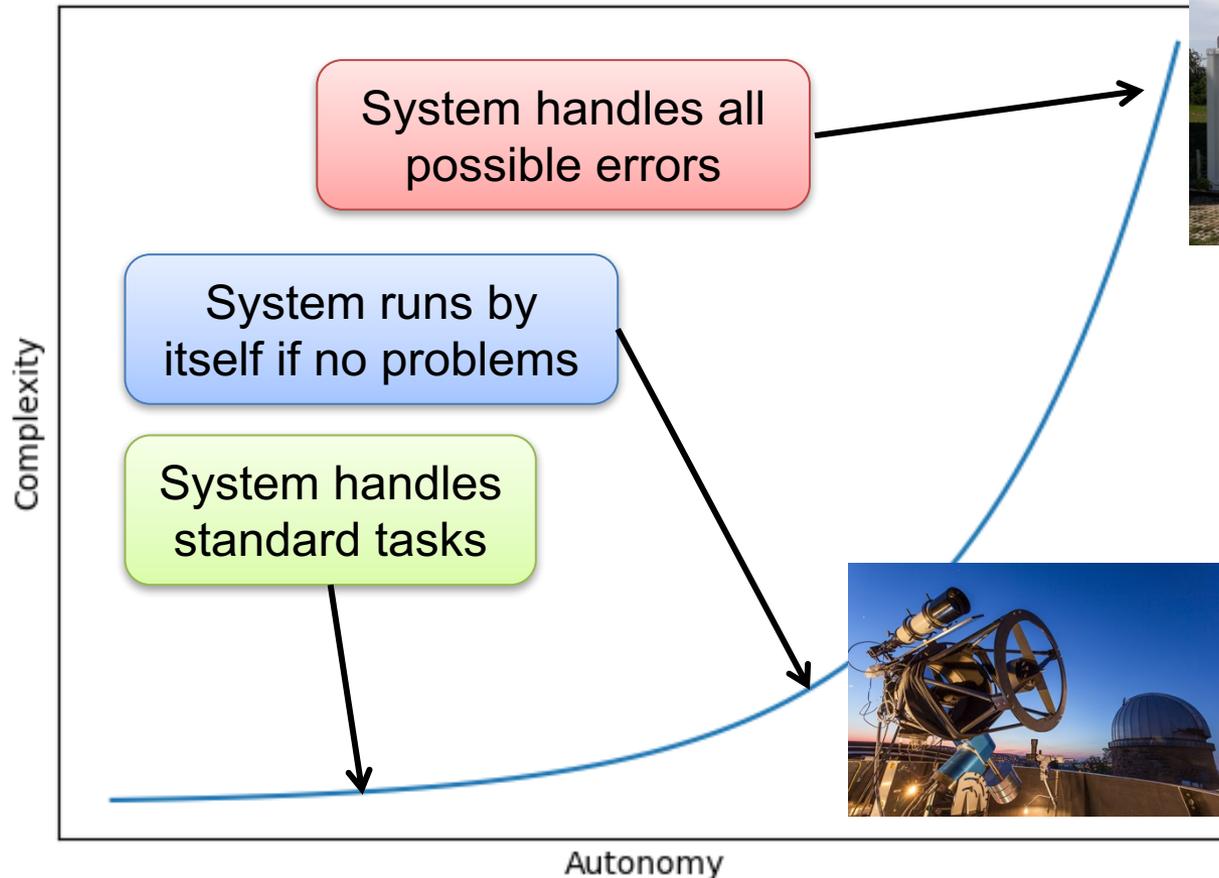
# Automatization: How much is useful?

- Can mean different things: Remote, robotic, autonomous, ...



# Automatization: How much is useful?

- Can mean different things: Remote, robotic, autonomous, ...



# Automatization with OOOS

- Separate programming from procedures
  - Better maintenance
  - Better testing (with simulation classes)
  - Flexibility

➔ Define procedures in XML

```
<run>
  <check class="hardware" method="cloud_sensor" on_problem="stop">
  <check class="hardware" method="all_sky_cam" on_problem="stop">
  <check class="hardware" method="detector" on_problem="restart_component">
  <check class="hardware" method="laser" on_problem="shutdown_system">
</run>

<stop>
  <task class="hardware" method="switch_roof" parameter="close" target_machin
  <task class="hardware" method="switch_mount" parameter="off" target_machine
</stop>
```



# Summary

- Hardware-independent control system offers
  - High flexibility
  - Fast hardware integration
  - Good reliability
- 100 kHz ranging is possible (and reasonable)



# Backup



# Stage 1 (“Robotic”)

- What for?
  - Safe budget on human observer crews
  - There is quick and easy access to the system
- Manual scheduling
- Handling all standard observation tasks, e.g.
  - Start tracking, trigger, event timer, open shutter
  - Fine-tune experiment parameters based on returns, e.g.
    - Laser-direction
    - Energy setting
- Fail-safe weather observation, roof close mechanism
- Remote signalling of errors, e.g. email or SMS



## Stage 2 (“Autonomous”)

- What for:
  - No easy access to the system (e.g. remote location)
- Automatic scheduling (incl. calibrations, etc.)
  - Possibly weather dependent
- Handle errors
  - restart hard- or software
  - Adjust settings
  - Continue running if non-essential hardware is missing
- Pre-process data and send it somewhere
- Complete remote access

